

Figure 1

1/7
AUS920010022US1

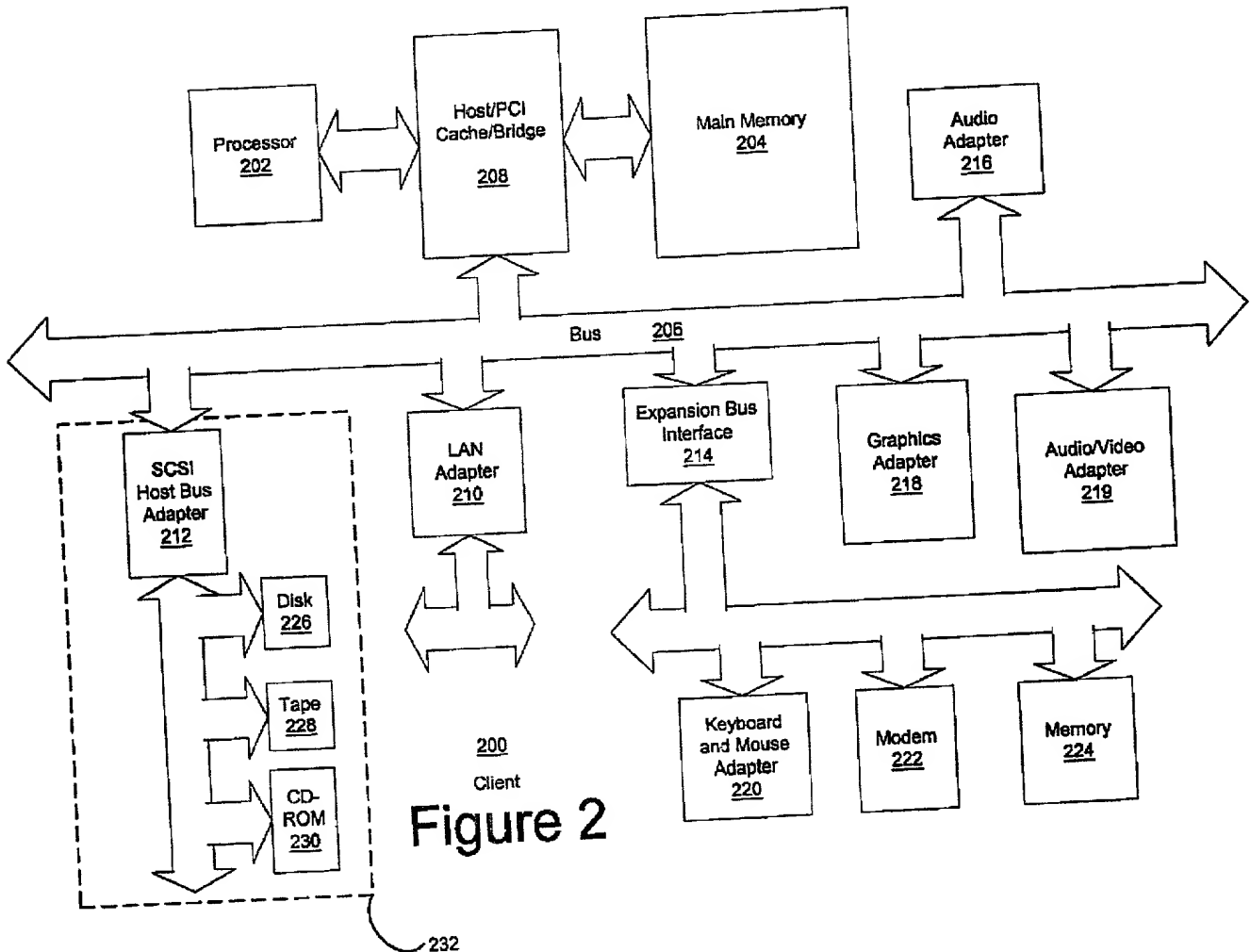
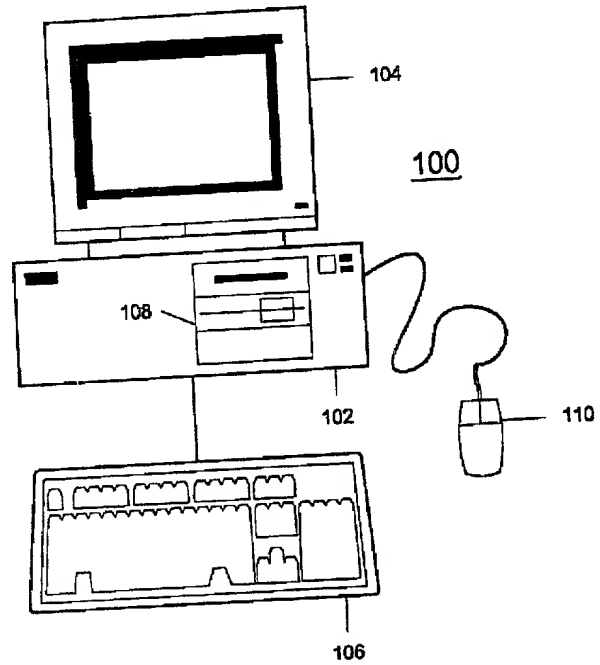


Figure 2

Figure 3

2/7
AUS920010022US1

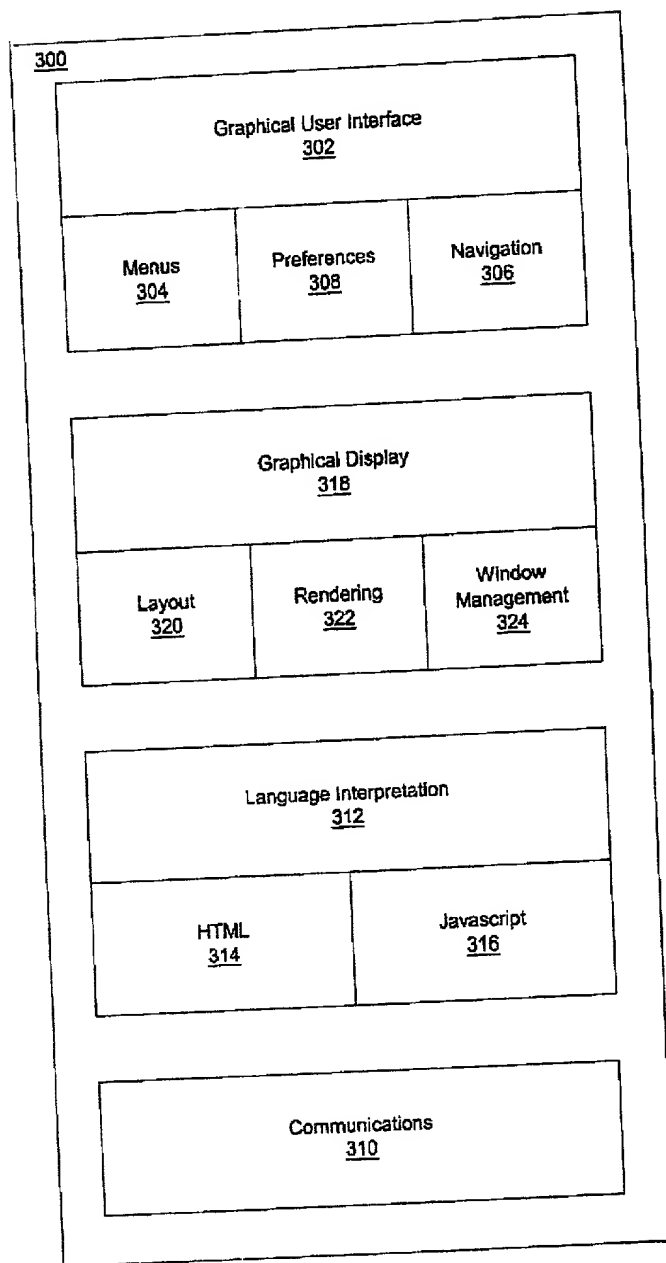
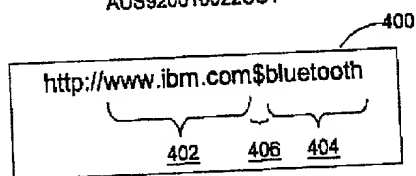


Figure 4

AUS920010022US1



Pub No: 2004/004560

Figure 5

3/7
AUS920010022US1

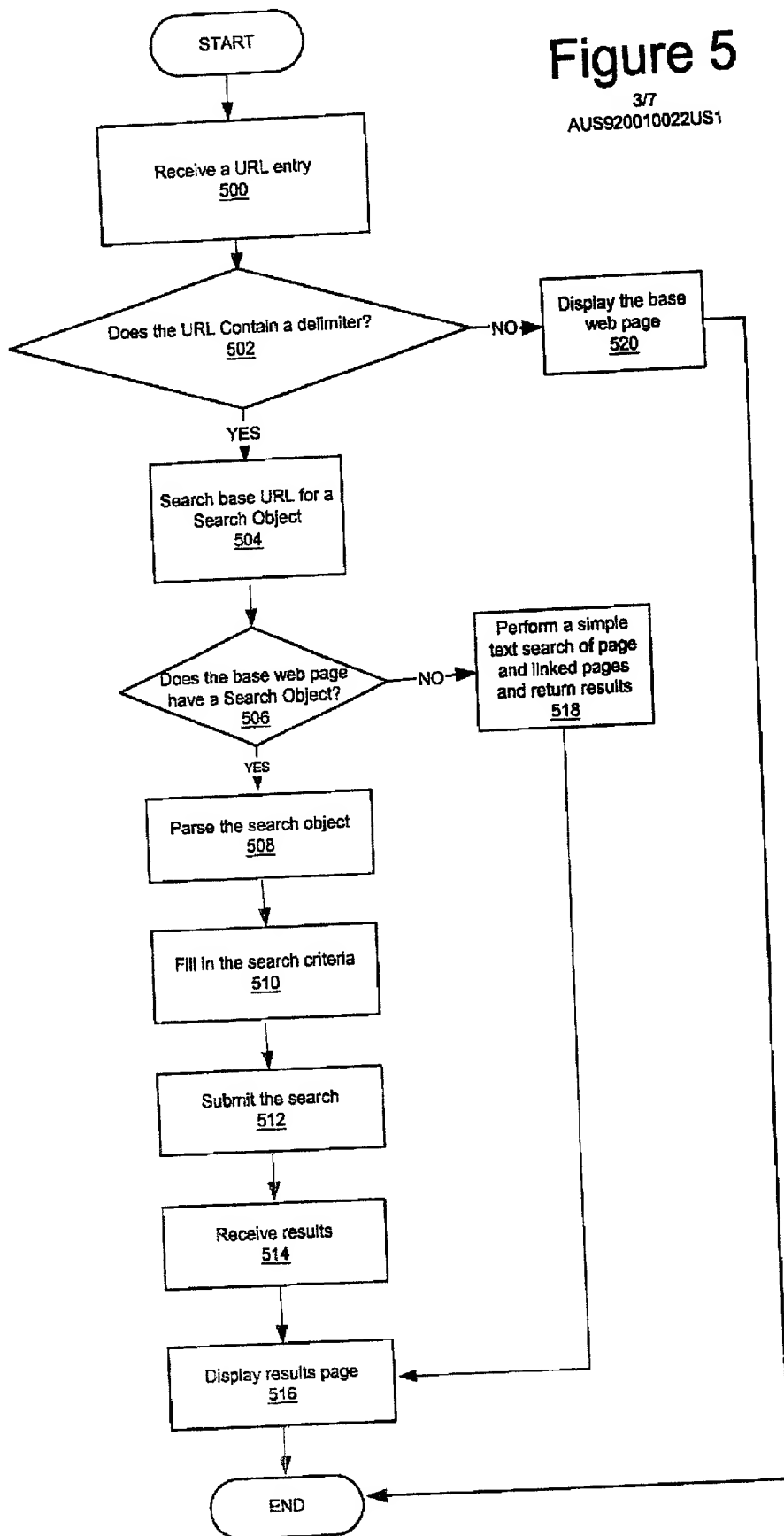


Figure 6A

4/7
AUS920010022US1

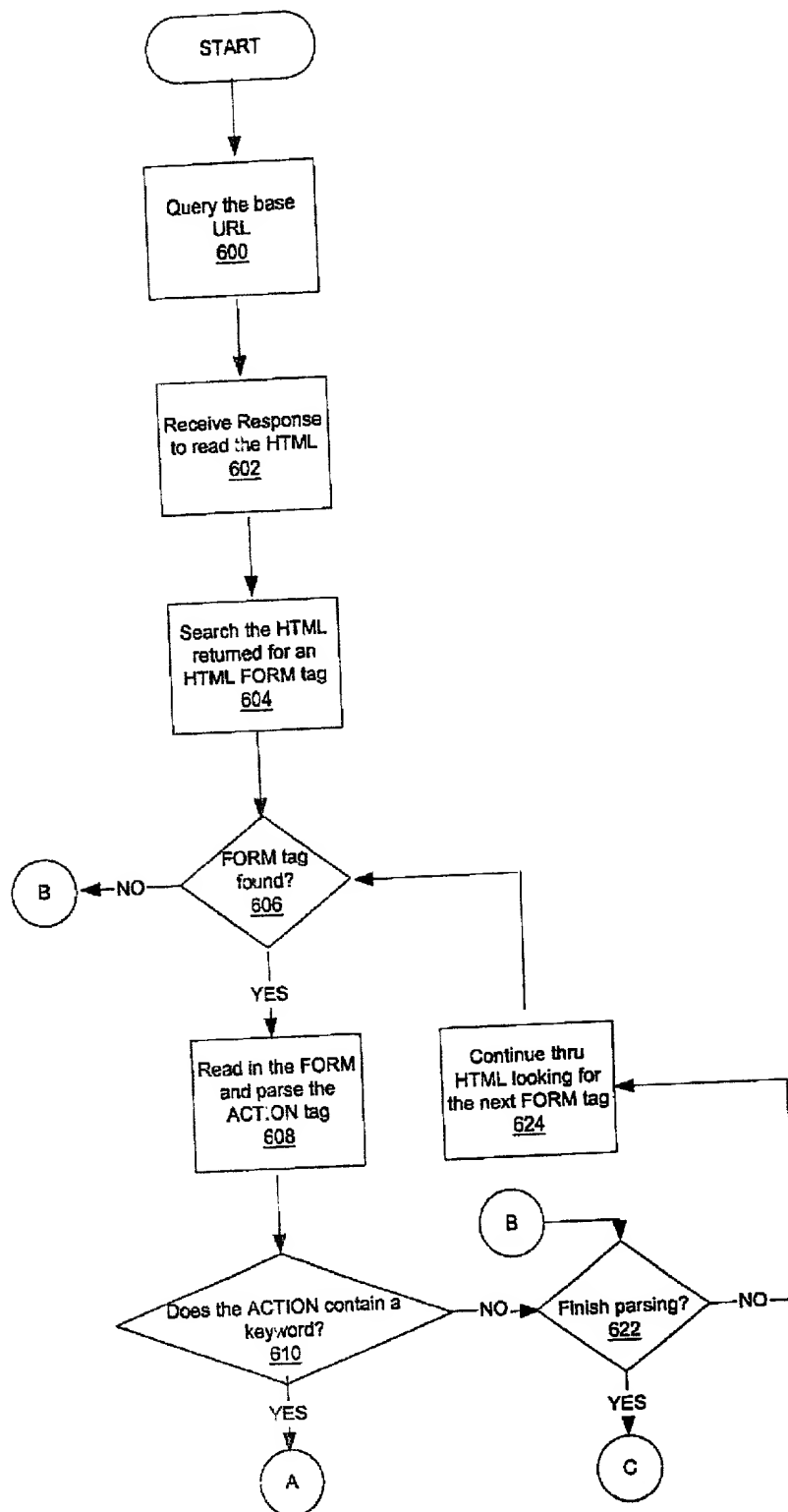


Figure 6B

5/7
AUS920010022US1

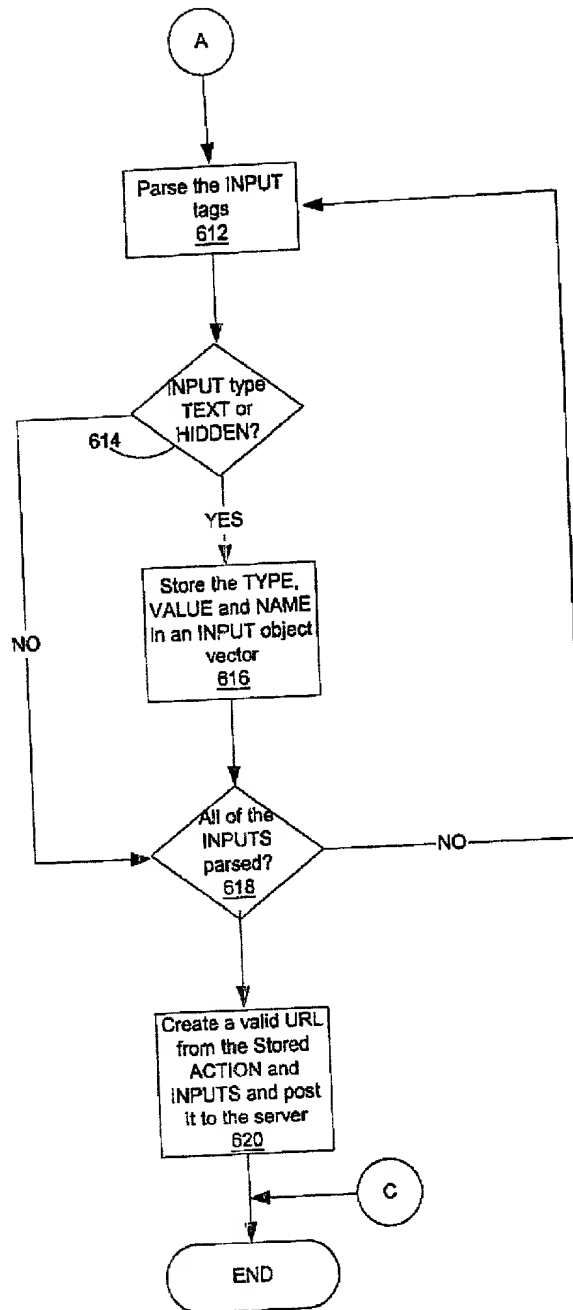


Figure 7A

6/7
AUS920010022US1

```
private String findSearchEngine(String sURL)
{
    String sSE = null;
    try
    {
        System.out.println("findSearchEngine: enter");

        // create a URL object from the http://xxx string
        URL tempURL = new URL(sURL);
        int index = 0;

        // Get a stream to read in the HTML from the page.
        BufferedReader br = new BufferedReader(new InputStreamReader(tempURL.openStream()));

        String line = br.readLine();
        while (line != null)
        {
            // Convert to lower case for case insensitive search
            line = line.toLowerCase();
            //System.out.println("findSearchEngine: Read in a line");
            //System.out.println("Line = " + line);

            // Need to make the assumption that the search engine action will
            // be inside of a form
            // So read in the entire form contents
            StringBuffer sbForm = new StringBuffer();
            if (line.indexOf("<form") >= 0)
            {
                System.out.println("findSearchEngine: found a form tag");
                while (line.indexOf("/form>") == -1)
                {
                    sbForm.append(line);
                    line = br.readLine().toLowerCase();
                } // end while

                // now that we have the entire form lets parse it for the action and
                // the inputs
                String sForm = sbForm.toString();
                System.out.println("findSearchEngine: Got a form the from is as follows:");
                System.out.println(sForm);
            }
        }
    }
}
```

[illegible]

AUS920010022US1

```
// Now parse the action of the form
setAction(doParseAction(sForm));
if (getAction().indexOf("search") != -1)
{
    // we have a candidate for the search so lets parse the inputs
    doParseInput(sForm);

    // now we have all that we need so lets create the search engine string
    Vector vecInputs = getInputs();
    if (vecInputs != null && vecInputs.size() > 0)
    {
        sSE = getAction().toString() + "?";
        if (sSE.startsWith("http:"))
        {
            sSE = sURL+sSE;
        } // end if
        HTMLInput htmlInput = null;
        Enumeration enum = vecInputs.elements();
        while (enum.hasMoreElements())
        {
            htmlInput = (HTMLInput)enum.nextElement();
            System.out.println("Input = " + htmlInput.toString());
            if (htmlInput.getType().equalsIgnoreCase("text"))
            {
                // this is probably the search word so add it in special
                sSE = sSE + htmlInput.getName() + "=" + getSearchKey() + "&";
            } // end if
            else
            {
                sSE = sSE + htmlInput.toString() + "&";
            } // end else
        } // end while
        // when we are all done remove the trailing &
        sSE = sSE.substring(0, sSE.length() - 1);
    } // end if
    // break out of the while loop because we have found the search engine.
    break;
} // end if
} // end if

line = br.readLine();
} // end while
}
catch (Exception e)
{
    e.printStackTrace();
} // end catch

System.out.println("findSearchEngine: exit");
return sSE;
} // end findSearchEngine
```